VŠB - Technická Univerzita Ostrava

Fakulta strojní

Katedra automatizační techniky a řízení

Aplikace RSS kanálů v prostředí univerzity

The RSS Technology Implementation in University Environment

Student: Martin Stříbný

Vedoucí bakalářské práce: Ing. Jiří Kulhánek, Ph.D.

Ostrava 2010

VŠB - Technická univerzita Ostrava
Fakulta strojní
Katedra automatizační techniky a řízení

Zadání bakalářské práce

Student: **Martin Stříbný**

Studijní program: B2341 Strojírenství

Studijní obor: 3902R001 Aplikovaná informatika a řízení

Téma: Aplikace RSS kanálů v prostředí univerzity
The RSS Technology Implementation in University Environment

Zásady pro vypracování:

1. Popište technologii RSS kanálů a její využití, porovnejte ji s jinými možnostmi.
2. Zmapujte webové informační zdroje na katedře, fakultě a univerzitě a vyhodnoťte, jaký typ informací by mohl být publikován formou RSS.
3. Realizujte sledování novinek na WWW serverech univerzity prostřednictvím RSS kanálů.
4. Realizujte agregaci zvolených existujících RSS kanálů univerzity.
5. Navrhněte další pokračování vytvořené aplikace.

Seznam doporučené odborné literatury:

[1] Holzner, S. Šindelář, J. *RSS : automatické doručování obsahu vašich WWW stránek.* 1. vyd. Brno: Computer Press, 2007, 278 s. ISBN 978-80-251-1479-7.
[2] Chow, S.W. *Programujeme mashup aplikace pro Web 2.0 v PHP.* 1.vyd. Brno: Computer Press, 2008, 280 s. ISBN 978-80-251-2057-6.
[3] LAVIN, P. *PHP - objektově orientované : koncepty, techniky a kód.* 1. vyd. Praha: GRADA Publishing. 2009, 211 s. ISBN 978-80-247-2137-8.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Jiří Kulhánek, Ph.D.**

Datum zadání: 18.12.2009
Datum odevzdání: 21.05.2010

prof. RNDr. Lubomír Smutný, Dr.
vedoucí katedry

prof. Ing. Radim Farana, CSc.
děkan fakulty

Místopřísežné prohlášení studenta

Prohlašuji, že jsem celou bakalářskou práci včetně příloh vypracoval samostatně pod vedením vedoucího bakalářské práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě dne 21. května 2010

…….

podpis studenta

Prohlašuji, že

- jsem byl seznámen s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121\2000 Sb., autorský zákon zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo.
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo nevýdělečně ke své vnitřní potřebě bakalářskou práci užít (§ 35 odst. 3.).
- souhlasím s tím, že bakalářská práce bude v elektronické podobě uložena v Ústřední knihovně VŠB-TUO k nahlédnutí a jeden výtisk bude uložen u vedoucího bakalářské práce. Souhlasím s tím, že údaje o kvalifikační práci budou zveřejněny v informačním systému VŠB-TUO.
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít toto dílo v rozsahu § 12 odst. 4 autorského zákona.
- bylo sjednáno, že užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- beru na vědomí, že odevzdáním své bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě: 21. května 2010

…….

podpis

Martin Stříbný
Bílá Bříza 458
Štěpánkovice, Svoboda 747 28

ANOTACE BAKALÁŘSKÉ PRÁCE

STŘÍBNÝ M. *Aplikace RSS kanálů v prostředí univerzity : bakalářská práce.* Ostrava : VŠB – Technická univerzita Ostrava, Fakulta strojní, Katedra automatizační techniky a řízení, 2010, 48 s. Vedoucí práce: Kulhánek J.

Hlavní část této bakalářské práce je zaměřena na tvorbu vyhledávacího robota, který dokáže nalézt změny na webových stránkách. Tyto změny se pak dále publikují pomocí RSS kanálů nebo přímo na webovém portálu k tomu určeném. V úvodu práce se zabývám technologií RSS kanálů a jejich využití v praxi. V druhé části jsou zmíněny webové stránky, ve kterých je možné vyhledávat změny, a ty které k tomu nejsou příliš vhodné. Dále je popsán samotný vyhledávací robot, technologie, způsob vyhledávání, indexace změn a další. Ve čtvrté části je popsán způsob agregace, prohledávání RSS kanálů a uložení nových stránek. Na závěr je popsán webový portál, který slouží k výpisu změn, jejich kategorizaci a také administraci samotného robota. Hlavním cílem této práce je umožnit uživatelům přístup k novinkám na webových stránkách, aniž by je museli pravidelně navštěvovat.

ANNOTATION OF BACHELOR THESIS

STŘÍBNÝ M. *The RSS Technology Implementation in University Environment*: *Bachelor Thesis.* VŠB – Technical University of Ostrava, Faculty of Mechanical Engineering, Department of Control Systems and Instrumentation, 2010, 48 p. Thesis head: Kulhanek J.

The main part of this thesis is focused on creating a search robot that can find changes on the website. These changes are then also published via RSS or directly to the web site for that purpose. The introduction of technology to deal with RSS feeds and their use in practice. The second part is mentioned website, where you can search for changes, and those who do not very useful. We also describe the actual search robot technology, way of searching, indexing and other changes. The fourth part describes the method of aggregation, scanning RSS feeds and save the new site. Finally, it describes the web site that is used to extract the changes, their categorization as well as the administration itself robot. The main objective of this work is to allow users access to news on the website without having to visit regularly.

Obsah

SEZNAM POUŽITÝCH ZKRATEK A ZNAČENÍ

ATOM	Atom Syndication Format
CDF	Channel Definition Format
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protokol
PHP	Hypertext Preprocesor (původně Personal Home Page)
RSS	Really Simple Syndication Rich Site Summary RDF Site Summary
SQL	Structured Query Langure
SGML	Standard Generalized Markup Language
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
WWW	World Wide Web
XHTML	extensible hypertext markup language
XML	Extensible Markup Langure

1 Úvod

Žijeme v moderní době počítačů a internetu, který umožňuje velmi snadné získávání informací. Jedním z možností, jak tyto informace získat, jsou také RSS kanály. Přitom mnoho uživatelů internetu ani netuší, co to RSS kanály jsou a ani jak je využít. Obzvláště, když jejich využití je více než snadné. Určitě každý někdy pracoval s nástrojem, který umožňuje sledování RSS kanálů a ani o tom třeba neví. Téměř všechny dnešní webové prohlížeče podporují jejich sledování. Portálové weby, typu seznam.cz, centrum.cz nebo igoogle.com, umožňují přidat na svou domovskou stránku vlastní obsah publikovaný právě v těchto RSS kanálech. Také sociální síť Facebook obsahuje aplikace umožňující jejich sledování. Problémem je, že o tom málokdo ví nebo nemá potřebu je využít.

Cílem této bakalářské práce je vytvořit robota, který bude procházet webové stránky Vysoké školy báňské, jejich fakult a jednotlivých kateder. Následně bude indexovat změny na těchto webových stránkách a vyhodnocovat jejich důležitost. Takto získané změny (novinky), pak bude možné sledovat jako novinky pomocí RSS kanálů nebo na webové stránce, kde se tyto novinky budou zobrazovat. Novinky jednotlivých webů bude možné mezi sebou kombinovat a vytvořit si tak svůj vlastní RSS kanál, kde bude vše, co Vás zajímá. Díky výše zmíněným možnostem sledování RSS kanálů budou tyto informace dostupné tam, kde je využijete nejvíce.

2 Co je RSS?

RSS je formát určený ke sledování novinek na webových stránkách neboli pro syndikaci obsahu. Nejčastěji se jedná o zpravodajské servery, blogy nebo jakékoli weby, které publikují nějaký obsah. RSS kanály nejčastěji obsahují jen titulky, odkazy a perexy jednotlivých článků. Hlavně proto, aby čtenář také navštívil web, na kterém je článek publikovaný a proměnil se tak v návštěvníka webu.

Zkratka RSS má 3 významy, které ale vyjadřují stejnou podstatu RSS kanálů.

1. Really Simple Syndication - skutečně jednoduchá syndikace
2. Rich Site Summary – Bohaté shrnutí webu
3. RDF Site Summary - RDF Shrnutí webu, kde RDF znamená Resource Description Format

[Wikipedia RSS 2010], [RSS 1.0 2010]

Všechny tři definice znamenají v podstatě to samé. Jde o formát, který umožňuje snadné shrnutí obsahu webových stránek. Takovéto shrnutí je pak možné využívat v RSS čtečkách a RSS agregátorech nebo i v jiných aplikacích a na webových portálech.

K čemu je to dobré?

V počátcích internetu bylo webových stránek jako šafránu a pokud jste se zajímali o nějaké téma, stačilo projít pár webů, (víc jich stejně nebylo) abyste se dozvěděli novinky v dané oblasti. S tím jak se začal internet zvětšovat, bylo zapotřebí procházet stále větší počet webů a v té době vznikla myšlenka syndikace obsahu.

Hlavní idea spočívá v tom, že informace putují k uživatelům a ne uživatelé za informacemi.

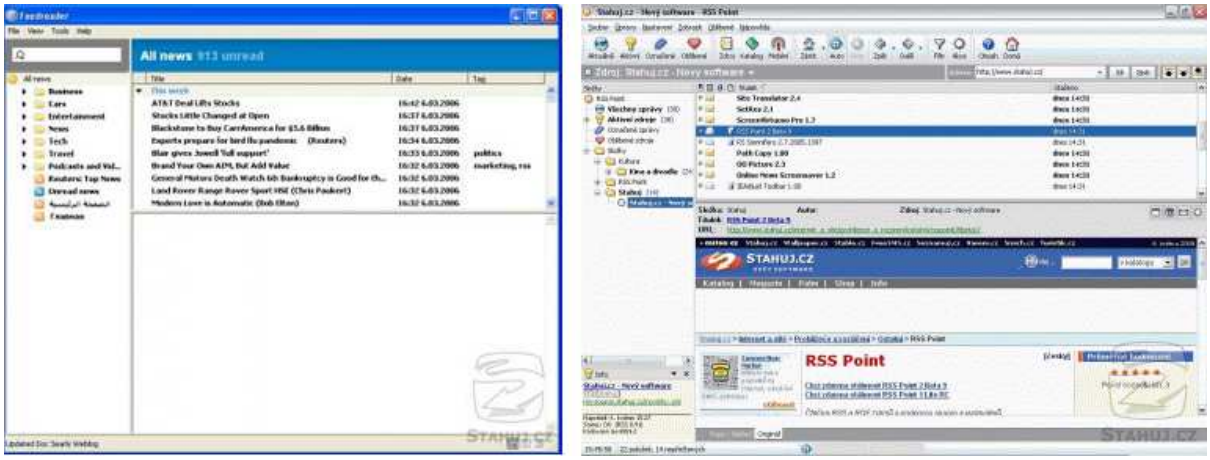
Díky používání RSS nemusíte obcházet všechny weby, které vás zajímají. Místo toho použijete RSS čtečku, ve které budete odebírat RSS kanály z vašich oblíbených zdrojů. A nové články budete sledovat na jednom místě. Teprve když Vás něco zaujme, pak přejdete na celý článek na daném webu.

2.1 Jak RSS používat

Již dříve jsem zmínil RSS čtečky, to však není jediná možnost použití a zároveň se i RSS čtečky dělí na 2 druhy.

Desktopové RSS čtečky (instalují se do počítače)

Desktopových čteček je k dispozici celá řada v nejrůznější kvalitě a ceně. Dnešní RSS čtečky jsou už zpravidla zdarma a volně ke stažení. Stačí si jen vybrat, která Vám nejvíc vyhovuje.

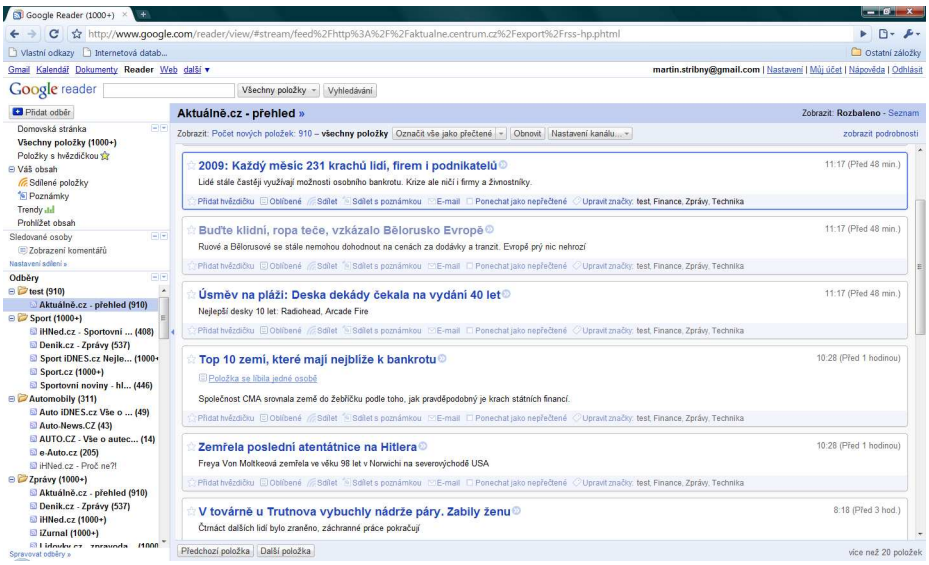


Obr. 1: Desktopové RSS čtečky FeedReader (vlevo) a RSS Point (vpravo)
(Převzato ze stahuj.cz)

Jistým druhem desktopové RSS čtečky jsou také samotné webové prohlížeče. Ve kterých je možné číst syndikovaný obsah tak, že si RSS zdroj přidáte do záložek/oblíbených a pak je čtete, jako běžené webové stránky.

Webové RSS čtečky (jsou na internetu dostupné online)

Zejména v poslední době jdou do popředí čtečky webové. Zejména z důvodu, že jsou dostupné z celého internetu a kdykoli jsou tak k dispozici. Díky tomu se můžete na nové články podívat např. v internetové kavárně, v práci, ve škole nebo doma a budete vědět, které jste už četli a které ještě ne. Zástupcem webových čteček je např. Google reader.



Obr. 2: RSS čtečka Google reader

RSS agregátory

Další způsob jak RSS využívat jsou RSS agregátory jako např. www.pravednes.cz, www.weblogy.cz aj. Tento způsob využití RSS je jednodušší pro koncové uživatele. Hlavně díky tomu, že agregátory jsou většinou tématicky zaměřené. Díky tomu nemusíte zdroje článku hledat vy sami, ale dělá to někdo jiný za vás. Vy si pak jen najdete téma, které Vás zajímá.

Portálové weby (seznam, centrum, igoogle apod.)

Portálové weby jsou někde na rozmezí webové RSS čtečky a RSS agregátoru. Na úvodní stránku si můžete přidávat vlastní RSS zdroje, které se vám zobrazí v boxech. Boxy jsou však omezené počtem zobrazených článků, pokud za den vyjde na některém webu víc článků, ke starším se už nedostanete. Navíc pokud máte hodně svých zdrojů, pak se brzy na úvodní stránce ztratíte, protože jeden RSS zdroj znamená jeden box na stránce. Také zde není žádná možnost kategorizace a zařazení obsahu. Tato možnost je určena spíše pro občasné čtenáře, kteří nemají příliš velké nároky.

Příjemnou kombinaci však vidím v zobrazování článků z RSS agregátorů na úvodní stránce portálových webů. V takovém případě totiž budou v jednom boxu články z celé kategorie agregátoru. Místo jednotlivých boxů pro každý zdroj zvlášť. Díky tomu pak budete mít na své domovské stránce články, které Vás zajímají, tím také padají všechny nevýhody spojené s kategorizací a velkým množstvím boxů, protože právě tyto 2 věci vyřeší agregační služba.

Obr. 3: RSS kanál z agregátoru weblogy.cz přidaný na seznam.cz

Proč používat RSS na webu (z pohledu webmastera)

Na první pohled by se mohlo zdát že, RSS kanály ubírají webovým stránkám návštěvníky. Do jisté míry je to pravda. Dochází však k posunu ke „kvalitním návštěvníkům“. Tzn., že stránku s daným článkem navštíví jen uživatel, který se o dané téma zajímá. Protože si už z popisku v RSS čtečce přečetl o čem článek je a podle toho se rozhodl, že web navštíví. Takže rozhodně nepřijdete o návštěvníky, které téma webu zajímá. A nakonec budou i vaši čtenáři rádi že se nemusí obtěžovat zbytečnou návštěvu webu, kde není nic nového. Místo toho si počkají, až bude skutečně co číst.

Použití RSS může dokonce zvýšit počet návštěv webu, zejména pokud publikujete často a hodně zajímavých článků. K tomuto efektu dochází hlavně kvůli tomu, že uživatelé daný článek vůbec nenajdou.

2.2 Historie a zařazení RSS

Historie RSS kanálů prošla bouřlivým vývojem, kdy jej spousta institucí odmítala, jako zbytečný, přes období, kdy se v RSS 1.0 snažili obsáhnout co možná největšího využití tohoto formátu, až zpátky ke konzervativnějšímu pojetí v RSS 2.0.

- 27.12.1997 – Byl navržený scriptingNews formát. Jediným rozdílem bylo, že měl RDF hlavičku, jinak to byla obyčejná variace XML formátu.
- 15.3.1999 – Firmou Netscape bylo navrženo RSS 0.90, pro použití v my.netscape.com, který rovněž podporuje scriptingNews formát.

- 15.6.1999 – Byl navržen scriptingNews 2.0b1, byl vylepšen a zahrnuje všechny funkce ve formátu RSS 0.90. Soukromě Netscape vyzval k přijetí vylepšení v tomto formátu, které nebyli přítomny ve formátu RSS 0.90.
- 10.7.1999 – Bylo navrženo RSS 0.91, speciálně Danem Libbym. Obsahuje většinu funkcí ze scriptingNews 2.0b1. Snaží se přejít k dalšímu standardnímu formátu a za tímto účelem je zahrnuto několik značek z populárního scriptingNews formátu. Hlavička RDF je pryč.
- 28.7.1999 – UserLand přijímá RSS 0.91 a scriptingNews formáty.
  Tým RSS v Netscape pomalu končí.
- 14.8.2000 – Byl publikovaný návrh RSS 1.0, který zpracovala soukromá skupina pod vedením Raela Dornfesta z O'Reilly. Je založená na RDF a použití jmenných prostorů. Většina prvků z předchozích formátů byla přestěhována do modulů. Stejně jako RSS 0.90 má hlavičku RDF, ale jinak je to zbrusu nový formát, který není spojen s žádnými předchozími formáty.
- 25.12. 2000 – Vzniklo RSS 0.92 což je RSS 0.91 s volitelnými elementy.
- 20.4.2001 – Diskutuje se o RSS 0.93, které nakonec nebylo nikdy nasazeno.
- 14.3.2002 – Vzniká MetaWeblog API, které spojuje RSS 0.92 s XML-RPC. To nabízí silné API pro blogy.
- 18.9.2002 – Vzniká RSS 2.0, což je RSS 0.92 s volitelnými elementy. API MetaWeblog bylo aktualizováno pro RSS 2.0. Ve vývoji byl tento formát nazvaný 0.94.
- 15.7.2003 – Specifikace RSS 2.0 byla propuštěna skrz Harvard pod licencí Creative Commons

[RSS History 2010]

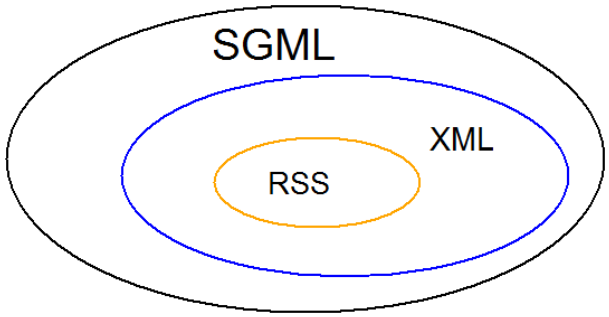
Zařazení RSS

RSS patří do rodiny formátů XML, z toho také plynou všechny jeho výhody i nevýhody.

XML je podmnožinou univerzálního značkovacího jazyka SGML

SGML (Standard Generalized Markup Language) je univerzální značkovací metajazyk, který umožňuje definovat značkovací jazyky jako své vlastní podmnožiny.

SGML je komplexní jazyk poskytující mnoho značkovacích syntaxí, ale jeho složitost brání většímu rozšíření.



Obr. 4: Zařazení RSS

2.3 Verze RSS

Historie RSS prošla bouřlivým vývojem, kdy zpočátku moc lidí nevěřilo masovému využití. Dosud není jasné, kdo vlastně vymyslel RSS a kdo je odpovědný za jeho vývoj. Kvůli tomu vznikly různé verze od různých autorů, které jsou někdy vzájemně nekompatibilní. Nicméně dnešní RSS čtečky nemají problém s žádným formátem.

Je zajímavé, že na webových serverech se stále drží několik verzí kanálu pro syndikaci od RSS 0.9 až po ATOM 1.0 i když naprostá většina RSS čteček zvládá všechny formáty. Navíc všechny formáty RSS jsou do značné míry zpětně kompatibilní.

RSS 0.9x

RSS 0.9x byla vytvořená společnostmi Netscape Communications a UserLand Software. Tato verze je do značné míry zkrácená verze jejich následovníků. Kouzlo této verze spočívá v její jednoduchosti. Obsahuje jen to, co je skutečně potřeba a nezatěžuje webmastery zbytečnými položkami.

Příklad zdroje v RSS 0.91

```
<?xml version="1.0" encoding="utf-8"?>
<rss version="0.91">
  <channel>
    <title>Titulek RSS kanálu.</title>
    <link>Absolutní url adresa webu.</link>
    <description>Popis RSS kanálu.</description>
    <language>Zkratka jazyka (cs, sk, en, …).</language>
    <image>
      <title>Titulek loga</title>
      <url>Absolutní adresa k obrázku.</url>
      <link>Odkaz, který bude reprezentovaný obrázkem.</link>
      <width>Šířka obrázku. (px)</width>
      <height>Výška obrázku. (px)</height>
      <description>Popis obrázku</description>
    </image>
    <item>
       <title>První položka</title>
       <link>http://www.web.cz/prvni.htm</link>
       <description>První příklad položky</description>
    </item>
    <item>
       <title>Druhá položka</title>
       <link>http://www.web.cz/druha.htm</link>
    </item>
  </channel>
</rss>
```

RSS 1.0

RSS ve verzi 1.0 je podstatně změněná oproti verzi 0.9x. Umožňuje definování více sémantických významů pro jednotlivé položky kanálu. A využívá více možností standartu RDF. Umožňuje snadnou rozšiřitelnost pomocí jmenných prostorů. Ale je komplikovanější než verze 0.9x a zpětně nekompatibilní.

Příklad zdroje v RSS 1.0

```
<?xml version="1.0" encoding="utf-8" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns="http://purl.org/rss/1.0/">

<channel rdf:about="http://www.web.cz">
  <title>Vzorové RSS 1.0</title>
  <link>http://www.web.cz/</link>
  <description>Příklad RSS kanálu 1.0.</description>
  <image rdf:resource="http://www.web.cz/logo.gif" />
  <items>
  <rdf:Seq>
    <rdf:li resource="http://www.web.cz/prvni.htm" />
    <rdf:li resource="http://www.web.cz/druha.htm" />
    </rdf:Seq>
  </items>
</channel>

<image rdf:about="http://www.web.cz/logo.gif">
  <url>http://www.web.cz/logo.gif</url>
  <link>http://www.web.cz/</link>
  <title>Popis obrázku</title>
</image>
<item rdf:about="http://www.web.cz/prvni.htm">
  <title>První položka</title>
  <link>http://www.web.cz/prvni.htm</link>
  <description>První příklad položky</description>
</item>
<item rdf:about="http://www.web.cz/druha.htm">
  <title>Druhá položka</title>
  <link>http://www.web.cz/druha.htm</link>
  <description></description>
</item>
</rdf:RDF>
```

RSS 2.0

Nejnovější verze z rodiny formátu RSS je zpětně kompatibilní z RSS 0.9x. Tzn., že přechod z nižší verze na vyšší, spočívá v podstatě ve změně čísla v hlavičce RSS kanálu. RSS 2.0 definuje nové elementy, které jsou využitelné v praxi. Jako např. lastBuildDate určující poslední změnu dokumentu, category, které určí kategorii článku, ttl určující dobu kešování aj.

Příklad zdroje v RSS 2.0

```
<?xml version="1.0"?>
<rss version="2.0">
  <channel>
    <title>Název kanálu</title>
    <link>http://www.web.cz/ </link>
    <description>Popis rss kanálu.</description>
    <language>cs</language>
    <pubDate>Tue, 11 Jun 2009 08:00:00 GMT</pubDate>
    <lastBuildDate>Tue, 11 Jun 2009 09:41:01 GMT</lastBuildDate>
    <docs>http://blogs.law.harvard.edu/tech/rss</docs>
    <generator>Název generátoru</generator>
    <managingEditor>email@web.cz</managingEditor>
    <webMaster>webmaster@web.cz </webMaster>
    <item>
      <title>První položka</title>
      <link>http://www.web.cz/prvni.htm</link>
      <description>Popis první položky.</description>
      <pubDate>Tue, 03 Jun 2003 09:39:21 GMT</pubDate>
      <guid>http://www.web.cz/prvni.htm </guid>
    </item>
    <item>
      <description>Popis druhé položky.</description>
      <pubDate>Fri, 30 May 2003 11:06:42 GMT</pubDate>
      <guid>http://www.web.cz/druha.htm</guid>
    </item>
    <item>
      <title>Třetí položka</title>
      <link>http://www.web.cz/treti.htm</link>
      <description>Popis třetí položky.</description>
      <pubDate>Fri, 30 May 2003 9:16:28 GMT</pubDate>
      <guid>http://www.web.cz/treti.htm</guid>
    </item>
  </channel>
</rss>
```

2.4 Další formáty pro syndikaci obsahu

CDF (Channel Definition Format) – Jde o zastaralý formát, který vyvinul Microsoft v roce 97. Záhy jej implementoval do svého webového prohlížeče IE 4. Tento formát se však vůbec neprosadil a nedočkal se tak širšího rozmachu.

ATOM (Atom Syndication Format) **–** Jedná se o nejnovější formát pro syndikaci obsahu. Je definován jako nástupce RSS, nicméně v dnešní době je stále více vidět formát RSS. Možná proto, že nové možnosti formátu ATOM jsou pro mnoho z uživatelů a webmastrů zbytečné.

Konkurenční Atom

Atom (plným názvem *Atom Syndication Format*) je webový standard pro publikování syndikovaného obsahu, přijatý IETF v prosinci 2005 jako RFC 4287. Je nástupcem formátu RSS. Kromě něj je pod RFC 5023 v říjnu 2007 přijat také *Atom Publishing Protocol* (zkráceně APP či AtomPub) umožňující vytváření a aktualizaci webových zdrojů ve formátu Atom pomocí HTTP.

Formát Atom také využívá značkovacího jazyka XML. To mají společné se všemi verzemi RSS. Má však mnohem více možností a více sémantických značek oproti staršímu RSS.

Rozdíly ATOMU oproti RSS

Kanál i jednotlivé položky musí mít uvedeny svůj název, jedinečný identifikátor (URI) a datum poslední změny. Povinné je rovněž jméno autora u každé položky, pokud není vyplněn element author obecné částí kanálu, který platí pro všechny. Zatímco u RSS existoval pouze jeden tag pro obsah článku description, který se využíval někdy pro souhrn z článku a někdy pro plný obsah, u Atomu jsou tyto elementy dva. Pro souhrn je vyčleněn element summary, zatímco pro plný obsah se používá content. V RSS nebylo možné zvolit, jaký formát obsahu je použit. Atom rozeznává např. prostý text, escapované HTML, XHTML, XML, binární obsah v kódování Base64 či odkaz na jiný webový zdroj. Existuje mnoho dalších rozdílů, které však nebyli pro potřeby této práce podstatné.

Příklad zdroje v Atomu

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom">
 <title>Example Feed</title>
 <subtitle>A subtitle.</subtitle>
 <link href="http://example.org/feed/" rel="self"/>
 <link href="http://example.org/"/>
 <updated>2003-12-13T18:30:02Z</updated>
 <author>
   <name>John Doe</name>
   <email>johndoe@example.com</email>
 </author>
 <id>urn:uuid:60a76c80-d399-11d9-b91C-0003939e0af6</id>
 <entry>
   <title>Atom-Powered Robots Run Amok</title>
   <link href="http://example.org/2003/12/13/atom03"/>
   <id>urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a</id>
   <updated>2003-12-13T18:30:02Z</updated>
   <summary>Some text.</summary>
 </entry>
</feed>
```

3 Mapování webových stránek

Vyhledávací robot je určen k indexování změn na statických stránkách. Jeho využití je nevhodné pro dynamický obsah jako jsou např. diskusní fóra, stránky s možností vkládání komentářů a stránky, které se často mění. Robot by často našel změnu obsahu a následně by byla zobrazena ve výpisu novinek, což by jej učinilo nepřehledným.

3.1 Weby vhodné k indexaci vyhledávacím robotem

Pro sledování novinek se v podstatě hodí jakékoli webové stránky statického rázu. V univerzitě tomu odpovídají prakticky všechny katedrální weby jednotlivých fakult. Některé katedry mají své vlastní webové stránky. U jiných jsou jejich prezentace součástí fakultních stránek. Tak jako tak, je možné sledovat fakulty jednotlivě jako samotné webové prezentace.

Stránky kateder jsou vesměs statické, jejich obsah se příliš nemění, a proto jsou pro je většina studentů ani nesleduje. Pokud dojde ke změně nějaké ze stránek nebo přibude stránka nová, většina lidí, kterých by se to mohlo týkat, to nezjistí. Z tohoto důvodu je vhodné nechat je sledovat vyhledávacím robotem, který bude agregovat všechny změny z webových stránek na jednom místě. Důsledkem toho pak bude, že informace budou snadněji dostupné pro studenty těchto kateder a fakult.

Například novinky na katedře Automatizační techniky a řízení jsou řešeny jednoduchou statickou stránkou, kde se postupně přidává nový obsah. Pokud tuto stránku nesledujete pravidelně, pak Vás pravděpodobně žádná novinka nezastihne včas.

Obr. 5: Novinky katedry Automatizační techniky a řízení

Dalším poměrně vhodným typem webu jsou osobní stránky některých pedagogů, kteří na svých stránkách umisťují novinky týkající se studia.

Jaký typ informací je vhodné sledovat?

Z předchozího textu je možné vydedukovat, že hlavním typem informací, které je dobré sledovat, jsou novinky na jednotlivých webech. Bohužel velké množství webů stránku s novinkami nemají. A weby, které stránku s novinkami obsahují, na ní nemusejí uvádět všechny informace a ani nesledovat její aktuálnost. Proto je vhodné sledovat všechny stránky.

Typy souboru, které jsou vhodné ke sledování, jsou v zásadě dva.

1. Běžné HTML stránky.
2. Dokumenty typu .doc, .pdf, .ppt apod.

Běžné HTML stránky

HTML stránka je v podstatě obyčejný textový soubor, který obsahuje značky pro vizuální prezentaci a sémantiku textu. HTML stránky umí robot procházet, nalézt v nich nadpisy nebo odstavce a z nich pak vytáhnout informace, které se budou zobrazovat ve výpise změn.

Dokumenty

Jiné dokumenty mohou sice obsahovat přínosný obsah pro uživatele, ovšem tyto soubory jsou pro robota nečitelné. Nemůže z nich načítat nadpis ani popisek a proto vychází pouze z názvu souboru, který je použit jako nadpis a popis zůstane prázdný. Jejich indexace je však vzhledem k možnosti, že obsahují nový obsah vhodná.

Nevhodné typy souboru pro sledování

Některé soubory je nevhodné a současně také zbytečné sledovat. Takovými soubory jsou například soubory typu videa nebo záznamy zvuku, případně obrázky. Jejich sledování je zbytečné zejména z důvodu, že pokud mají být dostupné, tak na ně musí vést odkaz z některé webové stránky. Pokud přibude na stránce odkaz, pak bude stránka zaindexovaná jako změněná, a bude zobrazena ve výpisu změn. Po přečtení této změněné stránky se také dostane na video (obrázek, zvuk), které na webu přibylo.

Druhým důvodem je také velikost těchto souborů. Jejich stažení trvá příliš dlouho a robot je stejně nedokáže zpracovat. Takže nadpis a popisek souboru bude stejně jako v předchozím případě převzat z názvu a popisek zůstane prázdný.

3.2 Weby nevhodné k indexaci

Weby, které obsahují různá fóra, ankety, komentáře se mohou měnit v zásadě z minuty na minuty a každá taková změna by způsobila indexaci robotem. Takovéto chování by v důsledku vedlo k znepřehlednění celého výpisu a zbytečnému ukládání víceméně duplicitních dat do databáze. Navíc systémy, které jsou dynamické, většinou obsahují vlastní RSS kanál, který je možné sledovat. Jeden příklad za všechny nevhodného použití je
E-learningový systém moodle .

E-learningový systém FS VŠB - vyuka.fs.vsb.cz

Na první pohled by se zdálo, že právě tento systém je ideální pro indexování změn tímto robotem. Nicméně je k tomu absolutně nevhodný. A to ze dvou důvodů.

1. Většina obsahu je chráněna heslem.
2. Na stránkách se vyskytuje dynamický obsah.

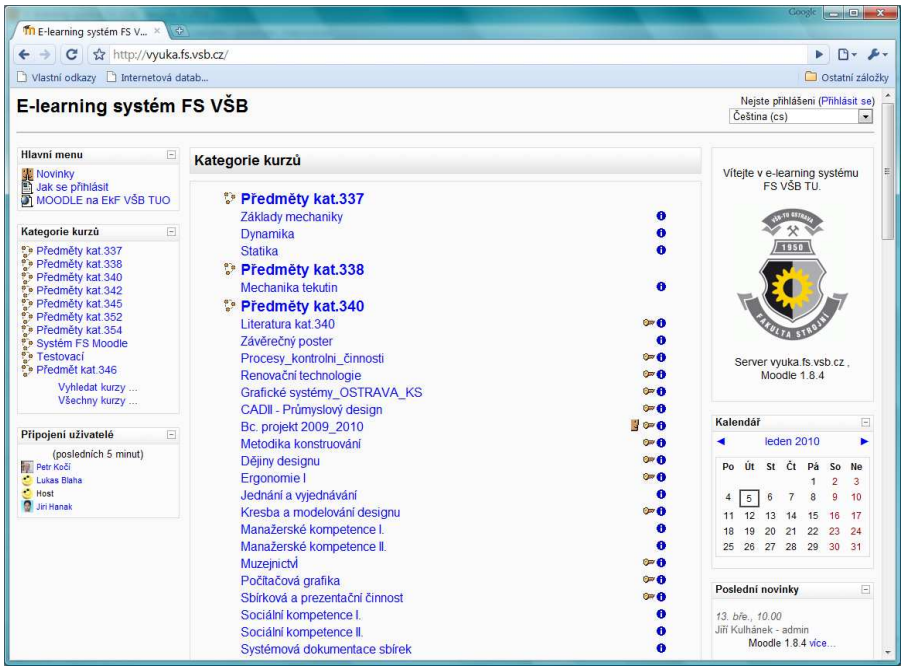
Ad. 1.

Obsah chráněný heslem by nemusel být pro robota nutně překážkou. Stačilo by zavést univerzální heslo pro přístup do systému Moodlu nebo robota upravit tak, aby mohl na určité stránky přistupovat se znalostí jejich hesla. Nicméně uživatele, kteří budou stahovat RSS kanály s novinkami nebo je sledovat přímo na webu, nebudou zajímat informace, ke kterým má přístup robot, ale jen informace, které zajímají právě je. Navíc pokud je výuka vedena s pomocí e-learningového systému, pak jsou většinou o změnách informováni přímo při výuce.

Ad. 2.

Na stránkách kurzů se mohou vyskytovat různé dynamické prvky jako jsou ankety, diskuse a samotné hodiny kurzu, které mohou být odkrývány postupně, podle postupu ve výuce.

Pro systém moodle by bylo daleko vhodnější použít RSS export novinek, který by mohl být následně agregován a zobrazen spolu s dalšími změnami.

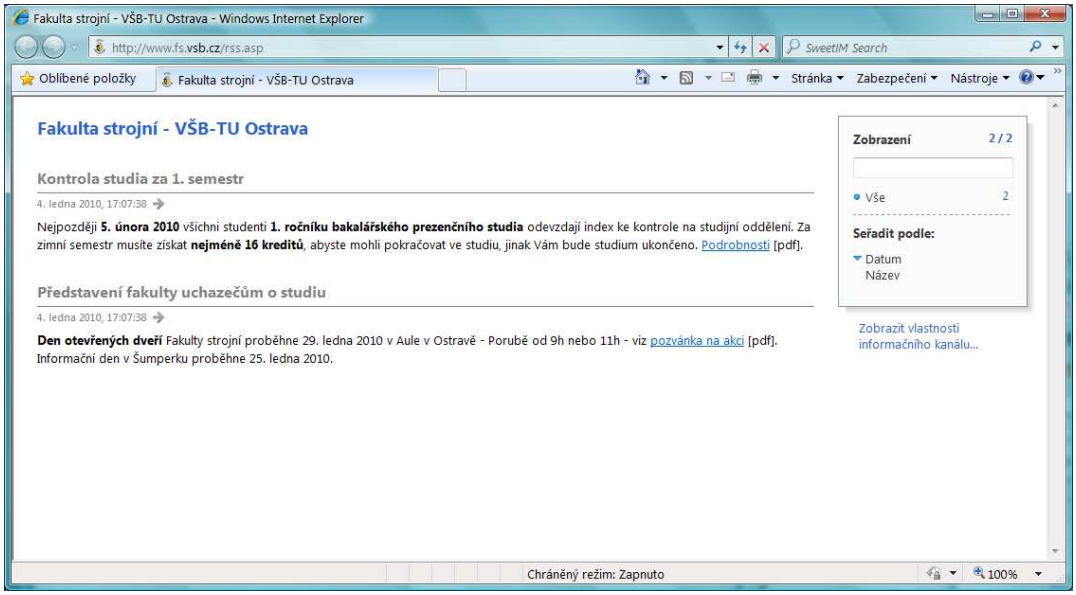


Obr. 6: Informační systém Moodle

3.3 Weby vhodné k agregaci

Agregací se myslí shrnutí více RSS kanálů do jednoho, tak aby se dali použít jako celek. Z toho vyplývá, že agregované mohou být jen ty weby, které obsahují zdroj RSS kanálu.

Tento požadavek splňují všechny fakultní weby, které zřejmě běží na jednotném systému. U fakult, jako jsou např. Ekonomická fakulta nebo fakulta stavební, kde je řada katedrálních prezentací vedena přímo pod fakultním webem, je pak také velice snadné sledovat novinky na všech těchto katedrách.



Obr. 7: RSS kanál webových stránek fakulty Strojní

Vlastní RSS kanál nemají jen fakultní weby, ale také další instituce a katedry, jako je např. katedra Informatiky na fakultě elektrotechniky a informatiky. Dalším vhodným příkladem může být institut geoinformatiky na fakultě hornicko-geologické, která má ne jeden RSS kanál, ale hned pět a také celá řada dalších webů.

4 Sledování novinek - vyhledávací robot

Robot je napsaný ve skriptovacím jazyku PHP a využívá databáze MySQL. Tato kombinace je široce podporovaná na různých webových serverech a je snadné jí provozovat.

4.1 Cíle robota

Vyhledávací robot prohledává všechny webové stránky a soubory na daných doménách a zjišťuje, zda-li nedošlo ke změně souboru. Pokud zjistí, že se soubor změnil, uloží tuto změnu do databáze a pokračuje v další činnosti. Nestará se už o samotnou publikaci změn.

Robot, jako takový, je určen k prohledávání statických stránek, na kterých nedochází k častým změnám. Takovéto weby nemají zpravidla pravidelné čtenáře a o změnách na webu se prakticky nikdo nedozví.

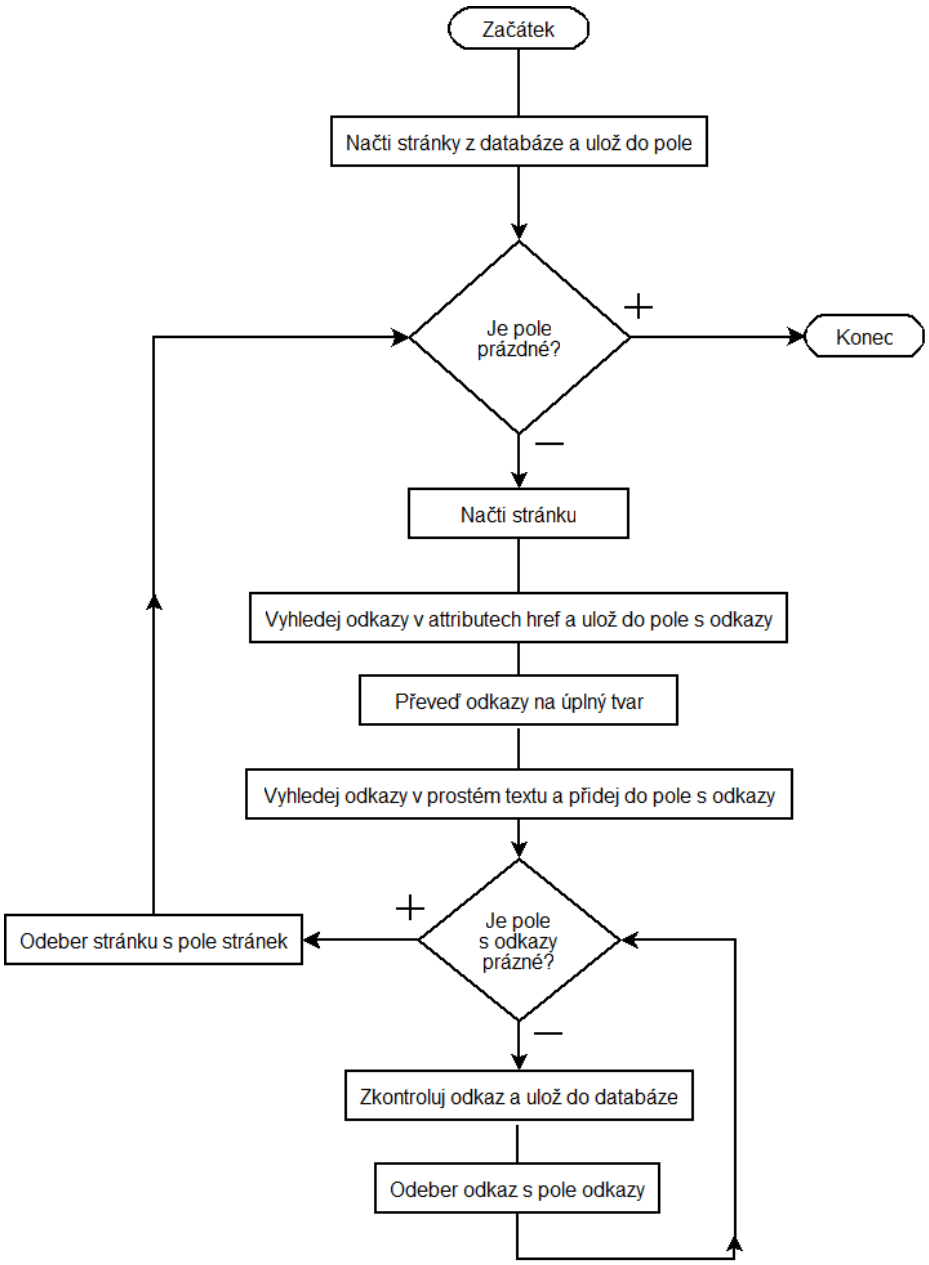
K čemu není robot určen?

Možná by se mohlo zdát, že je robot určený k vyhledávání na webových stránkách. Ale není tomu tak. Je sice pravda, že jsou stránky přečteny a zaindexovány, ale nejsou indexována žádná klíčová slova. Robot taky neobsahuje žádné mechanismy pro určení relevance a další důležité vlastnosti pro úspěšné vyhledávání na webu. Jediný cíl je zjistit změnu a určit důležitost této změny nikoli vhodnost stránky v závislosti na právě položeném vyhledávacím dotazu.

Vyhledávání

Robot nejprve potřebuje znát alespoň jednu adresu webové stránky, na které jsou umístěné další odkazy. Zpravidla je to úvodní stránka webu. Zaindexovány jsou všechny soubory, na které vede nějaký odkaz a robot se tak k nim může dostat. Adresy těchto odkazů jsou pak uloženy do stejné databáze, jako adresa původního souboru (Úvodní stránka). Díky tomu při příští kontrole robot narazí na nové soubory a opakuje se stejná činnost, jako s úvodní stránkou. Databáze souborů se pak exponenciálně zvětšuje, dokud nejsou nalezeny všechny soubory na dané doméně.

Postup vyhledávání je znázorněn na následujícím diagramu.



Obr. 8: Postup při vyhledávání nových odkazů

Soubor je vyhodnocený jako změněný, když nastala změna vůči předešlé indexaci v těchto bodech.

- Změnil se název souboru.
- Velikost souboru se neshoduje.
- Změnil se obsah souboru.
- Stránka vrátila hlavičku 301, 302 nebo 404.

Při ukládání změny se zároveň uloží relevance změny, která je popsána dále v textu. Díky tomu je následně možné odebírat jen ty změny, které jsou důležité.

Jak často se weby kontrolují a výpočet priority

Pokud by se při každém cyklu prohledávaly všechny adresy uložené v databázi, pak by byl robot neustále zahlcený prací. Takováto práce by byla z 99% zbytečná, protože žádný soubor by nebyl změněn. Proto je ke každému souboru přiřazena priorita. Díky, které je možné kontrolovat jen soubory, u kterých je pravděpodobné, že byly změněny. Při každém spuštění robota je pak vybráno jen několik souborů, které budou zkontrolovány.

Priorita je vypočítaná jako průměrný rozdíl času dvou po sobě jdoucích změn v minutách za určené období. Při výpočtu se také zohlední priorita, která je určena pro celý web. Může se stát, že se v daném období žádná změna nestala nebo že vypočítaná priorita je příliš velká. V tom případě se danému souboru přiřadí maximální priorita, která by měla odpovídat času 1-2 dnů. Maximální prioritu je možné určit v konfiguračním souboru robota.

Přičtením priority k datu poslední kontroly získáme datum budoucí kontroly. Podle tohoto data se následně seřadí všechny odkazy a první z nich se zkontrolují. Z toho vyplývá, že priorita je v podstatě čas v minutách, po jehož uplynutí bude stránka zkontrolována.

Pojem velikost priority

Z výpočtu priority vyplývá, že soubory s nejmenší prioritou jsou kontrolovány nejčastěji. Naopak soubory s největší prioritou jsou kontrolovány nejméně. V dalším textu budu používat pojem velikost priority ve smyslu tohoto výpočtu. Slovní spojení maximální velikost priority, tudíž nevyjadřuje soubory kontrolované nejčastěji, ale naopak soubory kontrolované nejméně.

Co vše se indexuje?

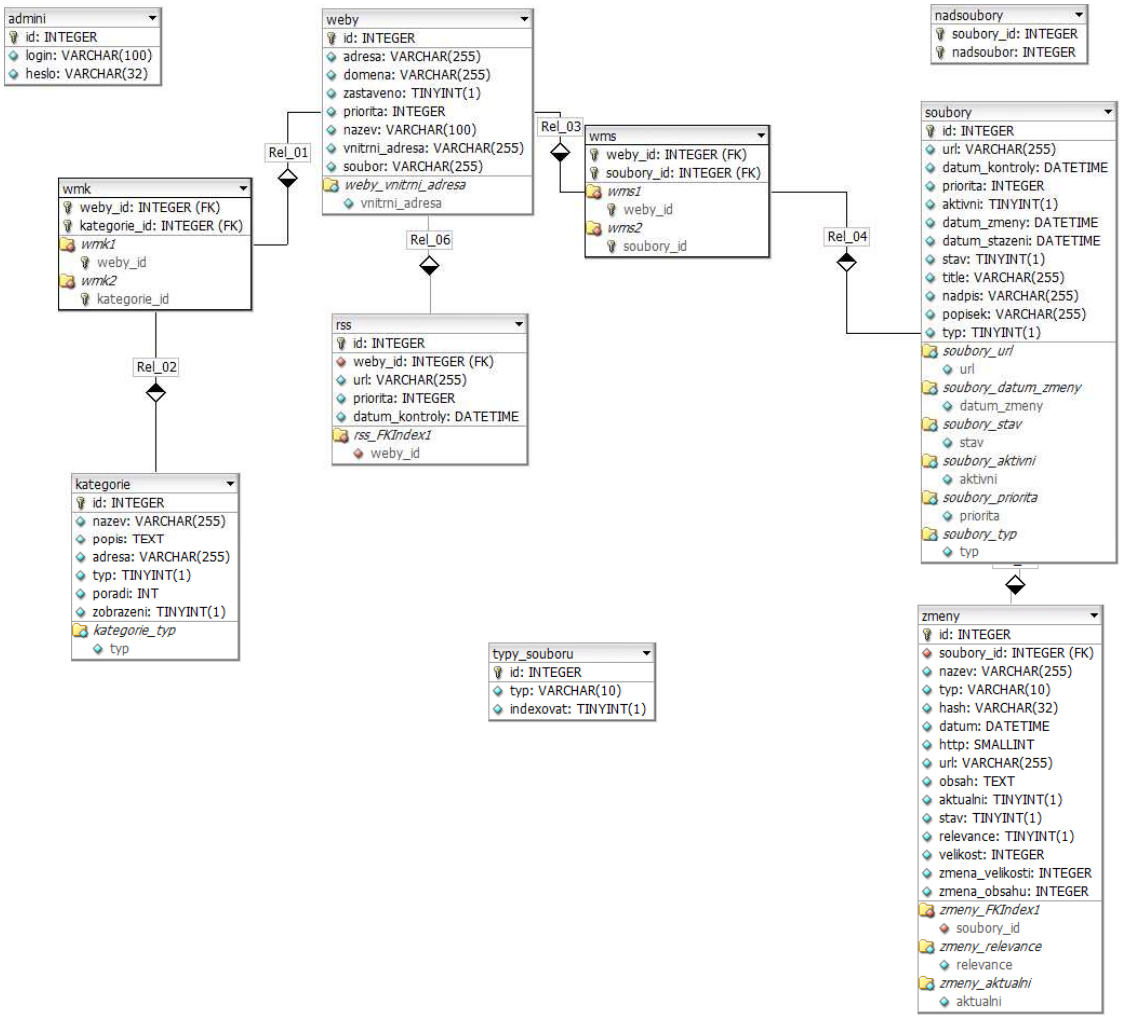
Robot není určen k prohledávání celého internetu, ale je omezen pouze na předem dané domény. Soubory na jiných doménách se robot nebude zabývat. Dále je možné omezit indexaci souborů podle jejich koncovky. Například soubory s koncovkami AVI, WMV, ASF, MOV apod. napovídají, že se jedná o video soubory, které není třeba indexovat.

Hlavičky 301, 302, 404

Vyhledávací robot také ukládá hlavičky odpovědí, které určují jeho chování. Při http odpovědi 302 se adresa souboru v databázi nezmění, ale uloží se informace o tom, že při poslední kontrole došlo k přesměrování. Při odpovědi 301 je situace trochu složitější. Informace o přesměrování se také uloží, ale zároveň dojde ke změně adresy souboru v databázi. Tuto změnu je pak možné publikovat jako změnu adresy souboru. Poslední možnost je odpověď 404, v tomto případě bude uložena informace o tom, že soubor nebyl nalezen a bude označen za neexistující. V tomto stavu zůstane do doby, dokud bude vracet hlavičku 404 a povede na něj nějaký odkaz.

V případě že na neexistující soubor vede odkaz, pak se robot zachová stejně, jakoby se jednalo o soubor, který se dlouho neměnil. Bude jej tedy pravidelně kontrolovat, s maximální prioritou pro případ, že by byl soubor obnoven.

4.2 Návrh databáze



Obr. 9: Návrh databáze

Základem databáze je tabulka weby, do které se ukládají všechny webové stránky, které se mají sledovat. Do jednotlivých řádků tabulky se ukládají následující informace.

- **adresa** – Url adresa na hlavní stránku webu (slouží pouze pro výpis změn).
- **domena** – Adresa webu, případně podadresáře (tuto adresu využívá robot).
- **zastaveno** – Příznak 1 nebo 0, pokud je web zastaven pak není sledován robotem.
- **nazev** – Název webu, který má být sledován (pro výpis).

Druhou důležitou tabulkou je tabulka soubory. Do této tabulky se ukládají veškeré zaindexované soubory, které robot našel. Z názvů řádků je patrné k jakému účelu slouží, proto zmíním jen ty, kde to nemusí být zřejmé.

- **url** – Adresa zaindexovaného souboru.
- **priorita** – Automaticky vypočítaná priorita, s jakou se bude soubor kontrolovat.
- **aktivni** – Příznak 1 nebo 0, pokud je 1, pak je soubor kontrolován, pokud 0, tak není.
- **title** – Obsah tagu <title> HTML stránky. Pokud se nejedná o HTML stránku, pak je prázdný.
- **stav** – Určuje, zda soubor existuje nebo ne, případně další stavy.
- **nadpis** – Pokud se jedná o HTML stránku pak je zde obsah prvního z tagů H1 - H3.
- **popisek** – Pokud jde o HTML stránku pak je zde obsah prvního tagu <p>.
- **typ** – Příznak, který určuje, zda byl soubor nalezen robotem nebo RSS kanálem.

Do tabulky zmeny se ukládá jakákoli změna daného souboru, která byla zaindexována robotem. Díky této tabulce je také možné sledovat historii změn jakékoli sledované webové stránky. Jednotlivé položky tabulky slouží k těmto účelům:

- **nazev –** Jméno zaindexovaného souboru.
- **typ –** Koncovka zaindexovaného souboru.
- **hash –** Kontrolní součet md5 obsahu souboru.
- **datum –** Datum zjištění změny.
- **http –** http odpověď serveru při stažení souboru.
- **url –** URL adresa souboru. Poslední známá URL adresa je také uložena v tabulce soubory.
- **obsah –** Celý obsah staženého souboru, který slouží pro výpočet změny.

- **aktuální –** Příznak 1 nebo 0, zda je tato změna poslední.
- **stav –** Stejně jako u souboru je zde příznak, zda soubor existuje nebo ne, případně další stavy.
- **relevance –** Příznak 1 nebo 0 zda je tato změna relevantní vůči předešlé (nemusí se kontrolovat při výpise, ale určí se při ukládání)
- **zmena_velikosti –** O kolik se zvětšil nebo zmenšil souboru vůči předešlé kontrole.
- **zmena_obsahu –** Vzdálenost Levenshtein mezi poslední a předposlední kontrolou.

V tabulce kategorie jsou uložené jednak veřejné kategorie (fakulty), které se zobrazují v postraním panelu na webové stránce, ale také jednotlivé kategorie z agregátoru. Je to proto, že se v zásadě řeší stejný úkol, jen se jednou zobrazí veřejně a podruhé soukromě. Tabulka obsahuje tyto řádky.

- **nazev –** Název kategorie, který bude zobrazen v menu.
- **popis –** Popis kategorie se využívá v tagu stránky <description>
- **adresa –** V případě veřejné kategorie je možné tuto adresu zvolit v administraci. V případě soukromé, je vygenerována automaticky z názvu.
- **typ –** Příznak, který určuje, zda se jedná o veřejnou nebo soukromou kategorii.
- **poradi –** Toto číslo určuje pořadí, v jakém budou kategorie seřazeny v menu.
- **zobrazeni –** Příznak 1 nebo 0 podle, zda se má kategorie v menu zobrazit.

Tabulky wmk a wms realizují vazbu M:N mezi jednotlivými tabulkami. Jejich názvy jsou odvozeny ze slov **W**eby **m**ají **K**ategorie a **W**eby **m**ají **S**oubory.

Tabulka typy_souboru slouží pro uložení typů souborů, které bude robot indexovat. Obsahuje jen 3 řádky: id, typ a indexovat. Id je automaticky generovaný primární klíč. Typ je v podstatě koncovka souboru ale může to být třeba odvozena i z mime hlavičky. A indexovat je příznak 1 nebo 0 určující, zda-li se má tato koncovka indexovat. V případě že koncovka v seznamu neexistuje, pak se defaultně neindexuje.

Předposlední tabulka se jmenuje nadsoubory. V této tabulce se buduje index stránek, které na sebe vzájemně odkazují. Prozatím toto není nikde využito. Jedná se spíš o možnost do budoucna, kdy by se dala vytvořit jakási interaktivní mapa všech oindexovaných souborů.

Poslední tabulkou, která stojí mimo vše ostatní je tabulka admini. V této tabulce jsou uloženy přihlašovací jména a hesla všech administrátorů, kteří mohou tento web spravovat.

4.3 Načítání webových stránek

Pro načtení webových stránek je použita knihovna CURL, která má na rozdíl od jiných variant širokou škálu možností a nastavení.

Při vrácení odpovědi 301 nebo 302 si robot zapamatuje, že nastala takováto situace a dojde k rekurznímu volání načítací funkce. Pokud dojde znovu k vrácení odpovědi 301 nebo 302 situace se opět opakuje, dokud nebude odpověď jiná nebo dokud nedojde k páté rekurzi. V takovém případě se bude se souborem zacházet stejně jako kdyby vrátil hlavičku 404. Během přesměrovávání se ukládají všechny odpovědi, které se cestou posbíraly. Ve výsledku je tedy možné, že odpovědi budou 3. (301, 302, 200 nebo 301, 302, 404 aj.)

Jaká odpověď bude nakonec uložena?

Jednoduchá odpověď zní vždy ta důležitější. Odpověď http 200 značí, že je vše v pořádku, ale z pohledu robota to není zajímavá informace. Důležitost se řadí podle následujícího seznamu.

1. http 404
2. http 302
3. http 301
4. http 200

Z toho vyplývá, že v případě http odpovědí 301, 302, 200 se uloží 301, protože soubor byl přesunut trvale. Naopak v případě http odpovědí 301, 302, 404 se uloží 404, protože soubor byl sice přesunut, ale nakonec bylo zjištěno, že byl také smazán, což je nejpodstatnější informace.

Vyhledávání odkazů a adres

Odkazy se získávají dvěma způsoby.

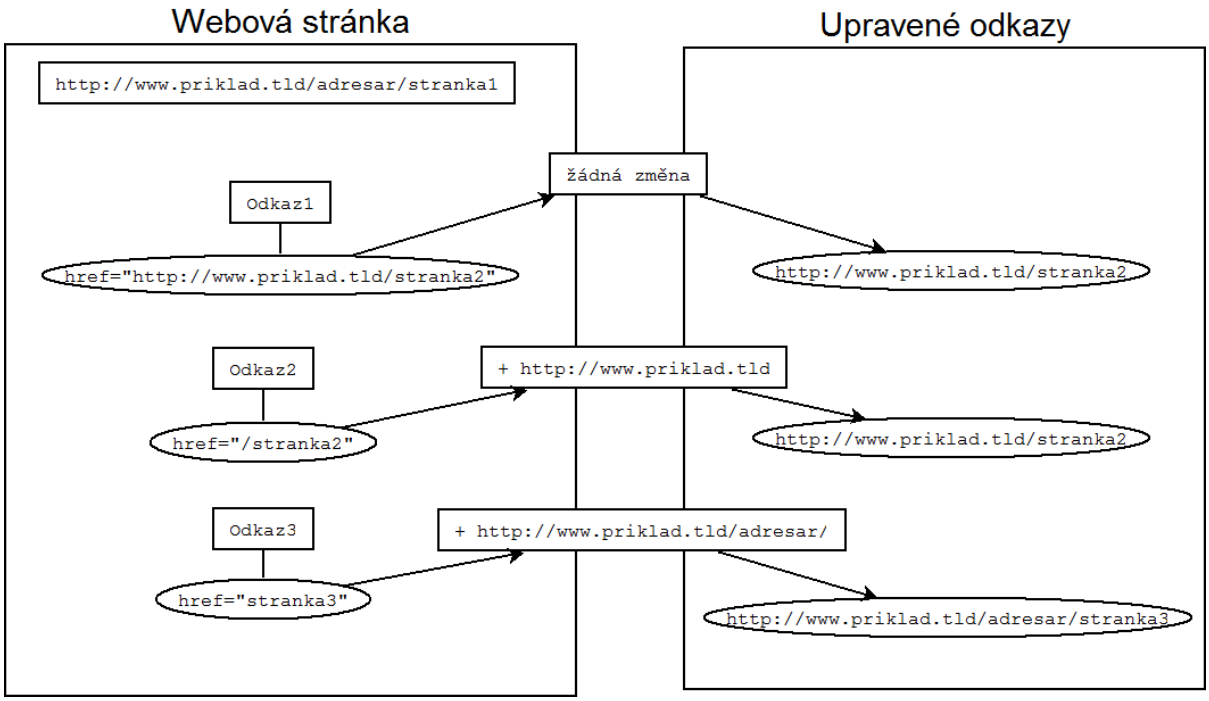
1. Nalezení všech adres z atributu href v definici odkazů.
2. Nalezení všech adres v prostém textu.

Ad. 1

V tomto případě je obvyklé, že adresy v odkazu nejsou kompletní. V takovém případě musí dojít k jejich doplnění podle adresy, na které se právě nachází. Doplnění adresy se řídí následujícími pravidly.

1. Pokud adresa začíná na některý z protokolů http:, http: nebo ftp: nedojde k žádné změně.
2. Pokud adresa začíná / (lomítkem) znamená to, že se soubor nachází na kořeni webu. Adresa tedy doplněna o protokol a hostname.
3. Pokud adresa začíná na cokoli jiného, bude doplněna o kompletní adresu aktuálního souboru až po poslední lomítko.

Jak se adresy doplňují, je zobrazeno v obrázku 10.



Obr. 10: Diagram doplňování url adres

Ad. 2

Adresy v textu musí mít úplný tvar, jinak robot nepozná, že se jedná o adresu. Adresy jsou vyhledávané regulárním výrazem:

```
 (http|https|ftp)\://([a-zA-Z0-9\.\-]+(\:[a-zA-Z0-9\.&%\$\-]+)*@)?((25[0-
5]|2[0-4][0-9]|[0-1]{1}[0-9]{2}|[1-9]{1}[0-9]{1}|[1-9])\.(25[0-5]|2[0-
4][0-9]|[0-1]{1}[0-9]{2}|[1-9]{1}[0-9]{1}|[1-9]|0)\.(25[0-5]|2[0-4][0-
9]|[0-1]{1}[0-9]{2}|[1-9]{1}[0-9]{1}|[1-9]|0)\.(25[0-5]|2[0-4][0-9]|[0-
1]{1}[0-9]{2}|[1-9]{1}[0-9]{1}|[0-9])|([a-zA-Z0-9\-]+\.)*[a-zA-Z0-9\-
]+\.[a-zA-Z]{2,4})(\:[0-9]+)?(/[^/][a-zA-Z0-9\.\,\?\'\\/\+&%\$#\=~_\-
@]*)*
```

Převzato z http://www.regularnivyrazy.info/url.html

Výraz je poměrně složitý, nicméně pokrývá snad skutečně všechny myslitelné URL. Akceptuje doménová jména, stejně jako zápis číselné IP adresy a zároveň pokrývá protokoly http, https a ftp.

4.4 Analýza získaných dat

Porovnání souborů

Stažený soubor je porovnáván vůči své předešlé variantě. Porovnávají se celkem 3 ukazatele, podle kterých je vyhodnoceno, zda-li se stránka změnila nebo ne.

1. Název souboru
2. http hlavičky
3. Změna obsahu souboru

První dva ukazatele jsou vyhodnoceny jednoduchým porovnáním. Zajímavější je pak třetí ukazatel, kterým je změna obsahu.

Změna obsahu

Prvním krokem je vypočítání kontrolního součtu pomocí funkce md5. Tento součet se porovná s předešlým a nakonec se také uloží do databáze. Pokud se součty shodují, pak nedošlo ke změně a proces je ukončen. Nicméně pokud se neshodují, pak je nutné provést další porovnání.

V dalším kroku se vypočítá, o kolik se od sebe tyto dvě verze liší. Výsledkem je číslo, které udává počet nutných změn k tomu, aby se oba soubory rovnaly. Každý soubor je nutné projít po bajtech, proto se tento postup provádí jen u souborů menších než 3Kb, kde jsou napřed odstraněny všechny HTML tagy. Větší soubory jsou kontrolovány pouze pomocí kontrolního součtu a porovnání velikosti souboru.

Pokud se kontroluje webová stránka, což bude nejčastější varianta, pak se robot pokusí načíst titulek, nadpis a popis stránky, které se pak použijí ve výpisu změn. Pokud se jedná o dokument, pak se místo nadpisu a titulku použije název souboru a popisek bude automaticky vygenerován.

Relevance změny

Ve výpisu změn je možné zvolit si jen výpis relevantních změn. Změna bude označena za relevantní, pokud splňuje následující body.

- Soubor je menší než 3Kb a počet změn je nejméně 20
- Soubor je větší než 3Kb a velikost souboru se změnila nejméně o 20 bajtů
- Soubor vrátil hlavičku 404 (soubor byl smazán)
- Soubor byl zaindexován robotem poprvé (soubor je na webu nový)

Každá uložená webová stránka je opět použita, jako zdroj nových odkazů.

4.5 Porovnávání obsahu (funkcí Levenshtein)

Vzdálenost Levenshtein je definována jako minimální počet změn, potřebných k tomu aby se jeden řetězec rovnal druhému. Jedna změna může být chápáno jako přidání, nahrazení nebo smazání některého znaku. [Wikipedia Levenshtein distance 2010]

Například Levenshtein vzdálenost mezi slovy pes a les je 1, protože dojde ke změně znaku p na l. Dalšími příklady mohou být například.

- Postel a Kostel = 1 – Dojde k nahrazení P a k.
- Bidlo a Šídlo = 2 – Dojde k nahrazení Bi a Ší.
- Polička a Přeslička = 4 – Dojte ke vložení znaků Př a nahrazení znaků Po za es. Případně obráceně dojde ke smazání znaků Př a nahrazení es za Po.

Příklad realizace funkce levenshtein

Běžně používané programovací algoritmy pro výpočet vzdálenosti levenshtein zahrnuje použití matice (n + 1) × (m + 1), kde m a n jsou délky porovnávaných řetězců. Tento algoritmus je založen na algoritmu Wagner-Fisher pro úpravy vzdálenosti.

Níže je pseudokód popisující tento algoritmus.

```
int LevenshteinDistance(char s[1..m], char t[1..n])
 {
   // d is a table with m+1 rows and n+1 columns
   declare int d[0..m, 0..n]

   for i from 0 to m
     d[i, 0] := i // pro mazání
   for j from 0 to n
     d[0, j] := j // pro vkládání

   for j from 1 to n
   {
     for i from 1 to m
     {
       if s[i] = t[j] then
```

```
        d[i, j] := d[i-1, j-1]
      else
        d[i, j] := minimum
                   (
                     d[i-1, j] + 1,  // mazání
                     d[i, j-1] + 1,  // vkládání
                     d[i-1, j-1] + 1 // nahrazení
                   )
    }
  }

  return d[n, m]
}
```

[Wikipedia Levenshtein distance 2010]

Levenshtein v PHP pracuje na úrovni bajtů, takže v případě kódování UTF může jedna změna písmene způsobit, že vzdálenost mezi řetězci je větší než ve skutečnosti. Nicméně tyto rozdíly jsou pro náš účel zanedbatelné.

Další metody porovnávání v PHP

Další možností při porovnávání obsahu je využít funkci **similar_text**, ta vrací počet stejných znaků ve dvou porovnávaných textech. V podstatě jde o podobnou úlohu, jako je výpočet vzdálenosti Levenshtein, nicméně je náročnější na výpočet.

Navíc platí že (délka textu) - (počet stejných znaků) = Levenshtein. Tato rovnice se dá také obrátit a zjistíme že (počet stejných znaků) = (délka textu) - Levenshtein. Proto je užití této funkce vcelku zbytečné.

Funkce založené na podobnosti mluveného textu

Tyto funkce uvádím pouze na okraj, protože se hodí spíše pro odhalování překlepů než k porovnávání změn v textu. Navíc jsou využitelné pouze pro anglický jazyk.

Tyto funkce jsou soundex a metaphone. Fungují na principu přiřazení stejného řetězce k různým podobně znějícím slovům. Tyto řetězce je pak možné jednoduše použít v podmínkách a zjistit jestli jsou si slova podobná.

5 Agregace RSS kanálů

Samotná agregace jiných RSS kanálu je v podstatě zjednodušená úloha vyhledávání změn na stránkách nebo v souborech. Zjednodušením proto, že vše co se objeví v RSS kanálu, lze považovat jako změnu stránky a není potřeba nic ověřovat.

Načítání RSS kanálů probíhá stejně, jako načítání webových stránek. Pro načtení je rovněž použita knihovna CURL, díky které je možné sledovat hlavičky odpovědi a snadno se realizuje rekurze při vrácení hlaviček ze skupiny 300.

RSS kanál je v podstatě obyčejný textový soubor, který je formátován pomocí značkovacího jazyka XML. Proto je nutné takovýto soubor upravit pro použití v programovacím jazyku. K tomu je využita jednoduchá knihovna v PHP s názvem simpleXML, ta slouží pro převod XML souboru do programových prvků typu objekt nebo pole, používaných v PHP. V takto naparsovaném souboru je pak už relativně snadné vyhledat informace, které potřebujeme.

Pro samotné uložení souboru se využívají stejné MySQL tabulky jako v případě nalezených stránek, protože se v podstatě jedná o stejný typ informací. Výsledkem obojího hledání je uložená HTML stránka. Ve stránce nalezené pomocí RSS kanálu je však navíc uložena informace o tom že pochází z RSS kanálu. Takto uložené soubory, pak mohou být používané jako nový zdroj odkazů pro sledování změn.

RSS a ATOM

V celém textu se věnuju popisu RSS kanálů. Tím ovšem nemám na mysli jen samotné RSS ale také kanály typu ATOM. Zároveň jsou tím myšleny také všechny verze RSS kanálů. RSS kanálem jsou tedy v dalším textu zamýšleny všechny formáty, které robot dokáže přečíst a zaindexovat.

5.1 Knihovna simpleXML

SimpleXML je knihovna pro PHP, která umožňuje snadnou a jednoduchou konverzi XML formátu do PHP objektů a polí. Knihovna simpleXML je knihovna, která se hodí spíše na jednoduché typy XML dokumentů. Nepodporuje všechny možnosti XML, jako jsou například jmenné prostory a další. Nicméně, pro naše účely zpracovaní RSS kanálů, je plně dostačující.

Dokumenty v XML mohou být načítány buď přímo ze souboru, nebo z řetězce. SimpleXML nám umožňuje pracovat pomocí moderního postupu objektově orientovaného programování. Základní princip je ten, že celý dokument je načtený do paměti do struktury objektů, jejichž jména odpovídají názvům elementů daného dokumentu.

Načtení dokumentu

Pro načtení dokumentu existují dvě funkce **simplexml_load_file()** nebo **simplexml_load_sring().** První z nich načítá data rovnou ze souboru, jehož adresa je funkci předána v parametru. Zatímco druhá funkce pracuje s řetězem, který je jí předán.

Pro názornost jsem použil jednoduchý příklad XML souboru.

```
$xml = simplexml_load_string(”<zamestnanec>
<jmeno>Karel</jmeno>
<prijmeni>Pospíšil</prijmeni>
</zamestnanec>”);
//pokud budu chtít vypsat jméno provedu toto:
echo $xml->jmeno;
//vypíše se Karel
```

Jako hlavní element se použije <zamestnanec>. Tento element je reprezentovaný přímo proměnou $xml. V jeho atributech pak budou všichni jeho potomci. Každý z elementů je po konverzi převeden na objekt, se kterým se dá následně snadno pracovat.

Čtení atributů

Pro čtení atributů elementů slouží funkce attributes(), která vrátí všechny atributy elementu, nad kterým je tato funkce volána ve formě pole. Více v příkladu níže.

```
$xml = simplexml_load_string('<zamestnanec jmeno=“Karel“
prijmeni=“Pospíšil“>
</zamestnanec>');
//pokud budu chtít vypsat jméno, budu postupovat následovně:
$atr = $xml->attributes(); // zavolám funkci attributes, která vrátí pole
echo $atr['jmeno']; //vypíše se Karel
echo $atr['prijmeni']; //vypíše se Pospíšil
```

Procházení více záznamů stejného jména.

Příklad:

```
<firma>
<zamestnanec>
<jmeno>Jiří</jmeno>
<prijmeni>Krátký</prijmeni>
</zamestnanec>
<zamestnanec>
<jmeno>Renata</jmeno>
<prijmeni>Nováková</prijmeni>
</zamestnanec>
</firma>
```

V tomto případě se jednotlivé elementy <zamestnanec> převedou na pole a jejich procházení vypadá následovně.

```
$cislo=0;
foreach($xml->zamestnanec as $zam){
$cislo++;
echo $cislo.". zaměstnanec se jmenuje: ".$zam->jmeno." ".$zam-
>prijmeni."<br/>";}
```

Výsledkem tohoto kódu bude výpis:

```
1. zaměstnanec se jmenuje: Jiří Krátký
2. zaměstnanec se jmenuje: Renata Nováková
```

V předchozím textu jsem prošel způsoby jak číst XML soubor pomocí knihovny simpleXML. Záměrně jsem se nevěnoval způsobů načtení kanálu ze souboru, tvorbě vlastního XML souboru a ani ukládání. Protože tyto možnosti nejsou v práci využity.

Vzorové příklady jsou převzaty z [Hrebenar Jiří]

5.2 Vyhledávací robot pro RSS

Pro sledování RSS kanálu je využit podobný princip, jako při sledování změn na webových stránkách. Sledování je opět realizováno pomocí vyhledávacího robota, který je spouštěn pravidelně cronem. Tento robot však neprochází jednotlivé webové stránky, ale pouze RSS kanály, které jsou předem určené.

Vyhledávání v RSS kanálu

Po té co je RSS soubor zpracován pomocí knihovny simpleXML, existuje v PHP jako objekt s poli, se kterým je možné pracovat standardními konstrukcemi jazyka, jako jsou podmínky cykly, porovnávací operátory a další.

V RSS se vyhledají všechny záznamy článků a následně jsou procházené cyklem. Každý záznam je porovnán s aktuálním stavem v databázi, a pokud se v kanálu nachází nový záznam, pak je uložen.

Uložení informací

Jak už bylo zmíněno dříve, položka nalezená v RSS kanálu reprezentuje reálnou webovou stránku, proto jsou také všechny záznamy uložené do stejných tabulek jako

v případě přímého nalezení stránky. Jediným rozdílem je že se navíc změní parametr typ, podle kterého je možné poznat, že se jedná o soubor nalezený v RSS kanálu.

Z RSS kanálu jsou uloženy následující položky.

- **title –** Při uložení bude použit ve sloupci title a nadpis.
- **link –** Bude použitý jako url adresa nalezené stránky.
- **description –** Použije se jako popisek při výpisu změn.

Údaje o čase jako datum změny, datum kontroly a datum stažení budou doplnění podle aktuálního času. Další údaje budou ponechány prázdné.

K záznamu vyhledanému pomocí RSS kanálu se již neukládá žádná změna, protože není nutné kontrolovat rozdíl oproti předchozí verzi. Toto chování vychází z předpokladu, že pokud dojde ke změně stránky, která byla takto nalezena, pak se také objeví nový záznam v RSS kanálu, díky kterému se pak aktualizuje předchozí varianta s novým časem.

Priorita stahování RSS kanálu

Pro sledování RSS kanálů je také využit systém priority, stejně jako při sledování webových stránek. Priorita zde však není přiřazena každé webové stránce, ale pouze RSS kanálu, který je sledován. Výpočet priority probíhá stejně, jako v předchozí kapitole, kdy se vychází z četnosti změn RSS kanálu v čase. RSS kanály, které jsou často aktualizované, se stahují častěji a naopak.

5.3 Formáty, které dokáže sledovat vyhledávací robot

Vyhledávací robot dokáže vyhledávat ve všech dnes běžně používaných formátech. Tím je myšleno RSS 0.9x, RSS 1, RSS 2 a samozřejmě také kanál ATOM. Tato univerzálnost byla poměrně snadno realizovatelná, protože jsou využity pouze základní části formátů (Titulek, Odkaz, Popis), které obsahují všechny formáty.

RSS

Všechny verze RSS kanálů obsahují základní data ve stejných značkách. Dokonce i formát RSS 1, který má však trochu složitější syntaxi, jelikož se adresy musejí uvádět na dvou místech současně. Po jednoduché transformaci je převeden do stejné struktury, jako mají ostatní verze RSS.

Při parsování kanálu RSS se provede pouze převod pomocí simpleXML a dále se přímo vychází se struktury, která touto transformací vznikne. Tím je myšlena struktura objektu, která je touto knihovnou vytvořena. A vzniká z názvu jednotlivých elementů. Na rozdíl od kanálu ATOM, který musí být převeden.

Malou výjimku tvoří RSS 1, která obsahuje elementy item přímo v rodičovském elementu. Po přesunutí všech elementů item pod element channel, se již ve všem ostatním shoduje s ostatními verzemi.

ATOM

Kanál ve formátu ATOM obsahuje stejné informace jako RSS, jen se nachází v jiných elementech. Kvůli tomu vznikne jiná struktura objektu po převodu pomocí simpleXML. Vzhledem k tomu že robot dále přímo využívá strukturu, která plyne z RSS kanálu, je nutné provést převod do správné podoby.

Struktura kanálu ATOM je hned po parsování převedena na kanál RSS. A to následujícím způsobem.

1. Element feed je převeden na element rss.
2. Vytvoří se element channel jako potomek elemetu rss.
3. Elementy entry jsou převedeny na elementy item a přidány do bloku channel.
4. Element sumary je převeden na element description.

Po této transformaci se struktury obou kanálu v základních částech shodují. A robot může dále s kanálem ATOM pracovat stejně jako by se jednalo o kanál RSS.

6 Webový portál

Součástí této bakalářské práce bylo vytvořit nejen samotné vyhledávání změn a agregaci RSS kanálů, ale také prostředek pro uživatele, kterým by tyto informace mohli sledovat. Proto byla vytvořena webová stránka, kde se tyto změny zobrazují. Tento web slouží nejen uživatelům, pro snadné nalezení novinek, ale jeho součástí je také administrace vyhledávacího robota.

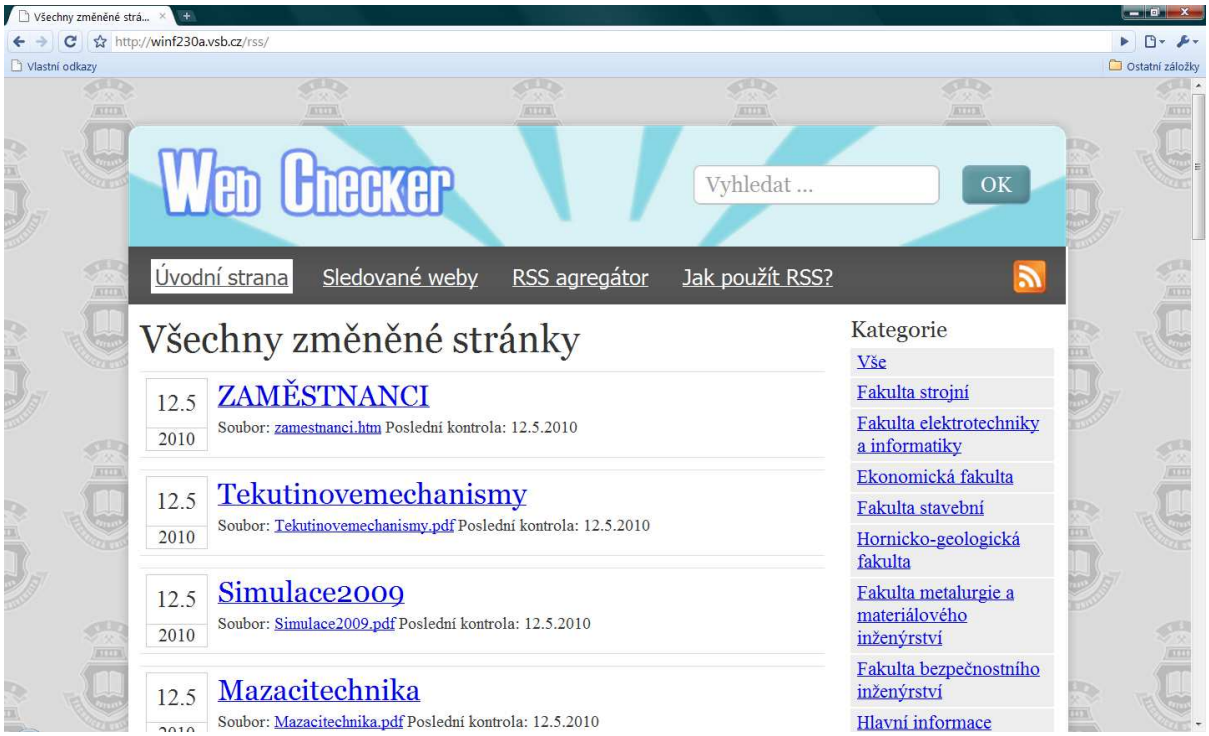
6.1 Ze strany uživatele

Pro využívání tohoto webu není třeba žádné registrace ani hledání vlastních zdrojů, které by se mohli hodit. Tento web slouží podobně jako běžné RSS agregátory, kde je předem připravený obsah, který by mohl uživatele zajímat.

Na rozdíl od běžných RSS agregátorů jsou zde zobrazené také změny, které se v RSS kanálech nenacházejí a jsou nalezené vyhledávacím robotem. Druhým rozdílem je možnost uživatele zvolit si jen určité weby, které ho zajímají a vytvořit si vlastní RSS kanál právě z těchto webů. Takovýto RSS kanál, pak může využít v jakékoli RSS čtečce, spolu s dalším obsahem.

Úvodní stránka a změněné soubory

Na úvodní stránce webu se zobrazují naposledy změněné soubory ze všech sledovaných webů. V pravém menu je pak možné vybrat si jen určitou fakultu, která Vás zajímá. A také deset naposledy změněných webů. Seznam všech sledovaných webů je k dispozici na stránce „Sledované weby“.



Obr. 11: Náhled úvodní strany webu

Vždy, když se uživatel nachází v nějakém výpise změn, je možné kliknout na ikonku RSS kanálu v horním menu a bude zobrazen RSS kanál právě prohlíženého výpisu. Na jakékoli jiné stránce se vždy zobrazí RSS kanál úvodní stránky. Kdykoli je také možné použít vyhledávání, které je dostupné na každé stránce.

RSS agregátor

Na stránce RSS agregátor, si může každý uživatel vybrat, jen ty stránky, které chce sledovat. Může si jednoduše zvolit celé fakulty nebo jen konkrétní weby, které ho zajímají.

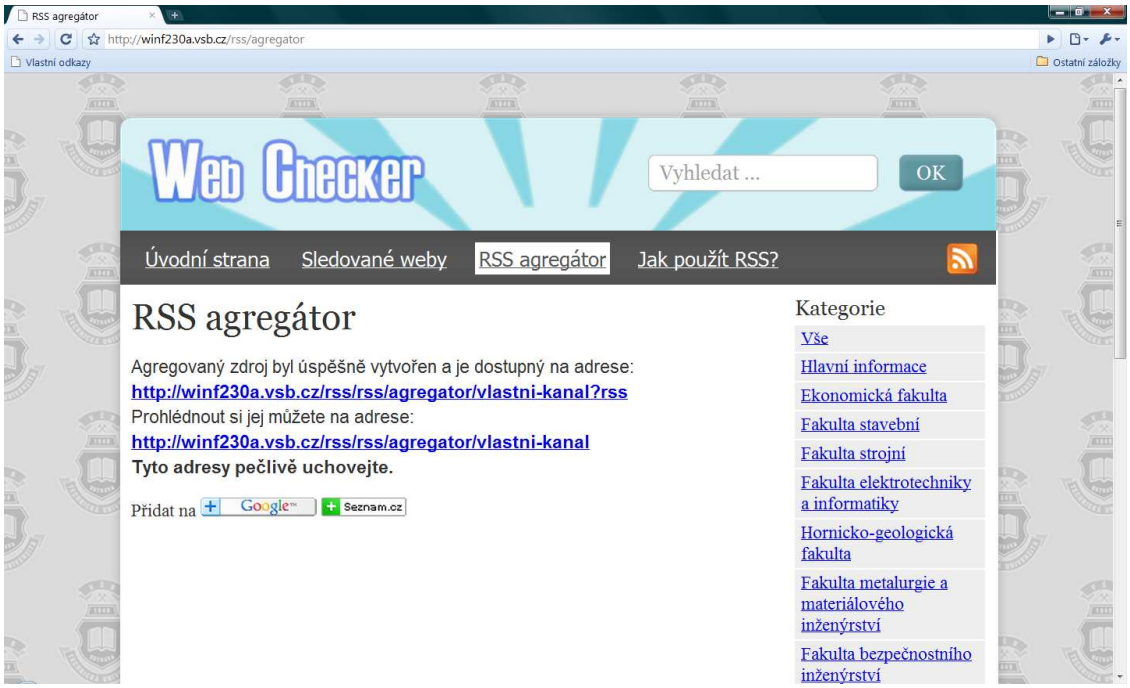
Pro snazší zapamatování si může také zvolit název svého RSS kanálu. Po potvrzení svého výběru bude vygenerován konkrétní RSS kanál, který bude dostupný pod adresou.

```
/agregator/{zvolený název kanálu}
```

Veškeré generování kanálů probíhá jednoduše bez registrace.

Takto vygenerovaný RSS kanál, je pak možné použít stejně jako jakýkoli jiný RSS kanál. Zároveň je možné zobrazit si tento agregovaný výstup tak jako je tomu na úvodní stránce. Adresa pro tuto variantu je:

```
/zmeny/agregator/{zvolený název kanálu}
```



Obr. 12: Náhled nově vytvořeného kanálu

Sledované weby a stránka „Jak použít RSS“

Na stránce sledované weby se nachází seznam všech webů, které jsou sledované robotem, nebo u kterých je sledován RSS kanál. Tato stránka slouží jen jako jednoduchý výpis pro uživatele, kteří chtějí vědět, jestli je jimi žádaný web sledován nebo ne. Kliknutím na vybraný zdroj si pak mohou zobrazit jen články z daného RSS kanálu.

Na stránce „Jak použít RSS kanály“, jak už z názvu napovídá, je výčet všech možností, jak používat RSS kanály. Stejné možnosti byli již zmíněné v první kapitole v této práci. Pro účely webu jsou pouze stručně shrnuty a vysvětleny jejich výhody a nevýhody.

6.2 Administrace

Administrace je dostupná na adrese /admin a je přístupná pouze administrátorům, kteří se musí prokázat uživatelským jménem a heslem. Také zde vede odkaz, který se nachází v patičce webu. Administrace slouží ke snadné správě vyhledávacího robota a webového portálu. Její jednotlivé části budou popsány níže.

Sledované weby

Na této stránce se nachází seznam všech sledovaných webů, které robot prochází. Je zde možné upravit prioritu sledování každého webu změnou čísla v sloupečku priorita. Čím nižší číslo se zde napíše, tím častěji budou soubory z daného webu procházené. Hned vedle se nachází zaškrtávací tlačítko Pozastaveno. Pokud je zaškrtnuto, pak tento web nebude sledován.

V dalším sloupci je zobrazena informace o tom zda daný web obsahuje sledováný RSS kanál. Po kliknutí na tento odkaz bude zobrazena detailnější úprava daného webu. Do stejné části se dostanete kliknutím na odkaz „upravit/smazat“.

Přidání nového webu ke sledování

Po kliknutí na odkaz „Přidat nový web“ se zobrazí formulář pro vložení nové domény. Pro přidání webu je nutné vyplnit následující položky.

- **Název webu –** Tento název se bude zobrazovat v postraním menu nebo na stránce RSS agregátor.

- **Adresa webu –** Adresa, po jejíž kliknutí se zobrazí sledovaný web. (Slouží pouze pro uživatele, ne pro robota.)

- **Postfix domény –** Adresa, která slouží robotovi pro prohledávání webů. Zapisuje se bez protokolu a může obsahovat i podadresáře. Tento údaj možná zní na první pohled nesmyslně ale je zde proto, že důležitost adresy roste zleva doprava. Například při sledování stránky robota, které se nachází na adrese **http://winf230a.vsb.cz/rss/** je třeba zadat tuto adresu **winf230a.vsb.cz/rss/**. Pokud bych chtěl sledovat všechny weby na adrese **http://winf230a.vsb.cz** je třeba zadat **winf230a.vsb.cz**. A konečně pokud bych chtěl sledovat všechny weby na doméně VŠB, pak je třeba zadat **vsb.cz**

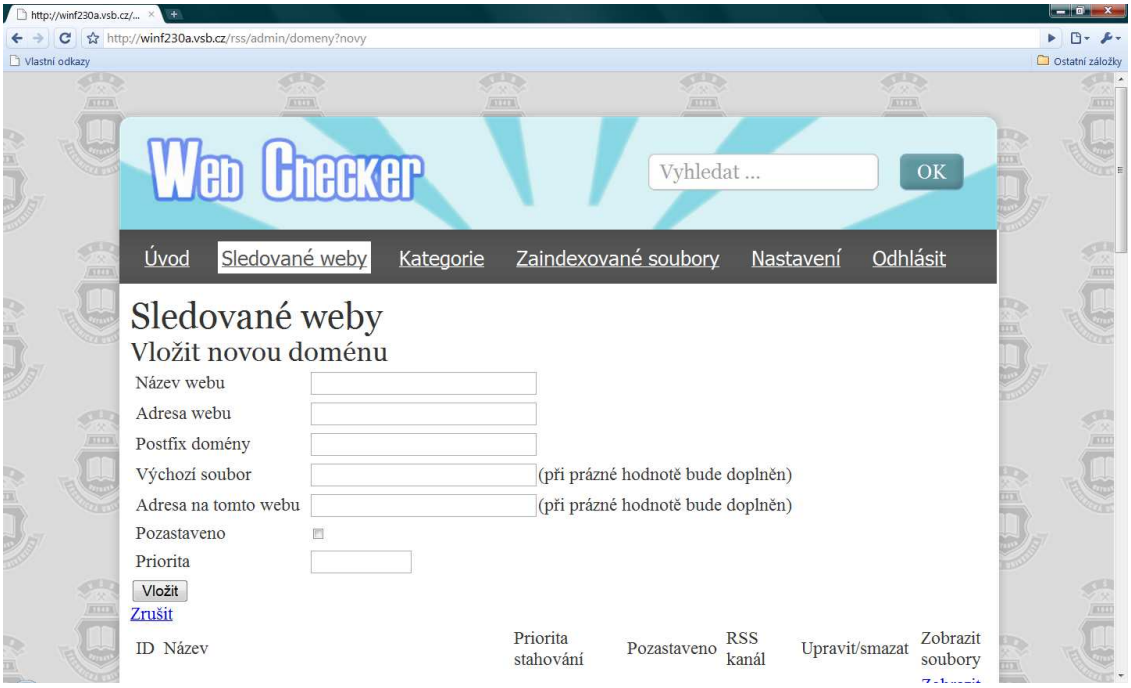
- **Výchozí soubor –** Adresa nějakého souboru na doméně, která se má sledovat. Pro většinu případů se bude tato adresa shodovat s postfixem, pouze s přidaným protokolem, proto se tento údaj nemusí vyplňovat.

- **Adresa na tomto webu –** Jak z názvu vyplývá, tak tato adresa bude použita ve všech odkazech, které povedou na výpis z tohoto webu. Nejedná se o klasickou http adresu, jde v podstatě o název webu bez háčků čárek a mezer, který bude využit pro tvorbu celé adresy

- **Pozastaveno –** Stejně tak jako ve výpise sledovaných webů je i zde možné zaškrtnutím tohoto tlačítka možné pozastavit sledování daného webu.

- **Priorita –** Údaj priorita má také stejný význam jako ve výpise sledovaných webů a v zásadě udává časový interval v minutách, kdy se bude robot na tento web vracet.

Po vložení nového webu bude zobrazena stránka pro editaci.



Obr. 13: Vložení nového webu ke sledování

Editace sledovaného webu

Na začátku stránky pro editaci se nachází stejný formulář jako při vkládání nového webu. Zde je možné jakékoli údaje změnit. Níže jsou pak zobrazené kategorie, do kterých je web zařazen. Tyto kategorie je možné odebrat zaškrtnutím pole zrušit a následným uložením změn.

Ihned po vložení nového webu je nutné přiřadit jej pod některou s kategorií, které jsou dostupné ve třech seznamech. Tři jsou zde pouze pro snadný výběr více kategorií současně. Pokud web zapadá pouze do jedné kategorie, pak stačí vybrat jedna. Pokud se hodí do více než tří kategorií, pak je nutné vybrat si nejprve tři kategorie, potvrdit výběr a následně přidat další.

Dále se na této stránce zobrazuje seznam RSS kanálů, které jsou k danému webu přiřazené. Tyto kanály je také možné stejně jako kategorie odstranit zaškrtnutím pole zrušit a uložením změny. Nové RSS kanály je možné přiřadit pomocí textového pole níže, kde je nutné zadat přesnou adresu RSS kanálu. Více RSS kanálu je nutné přidat postupně.

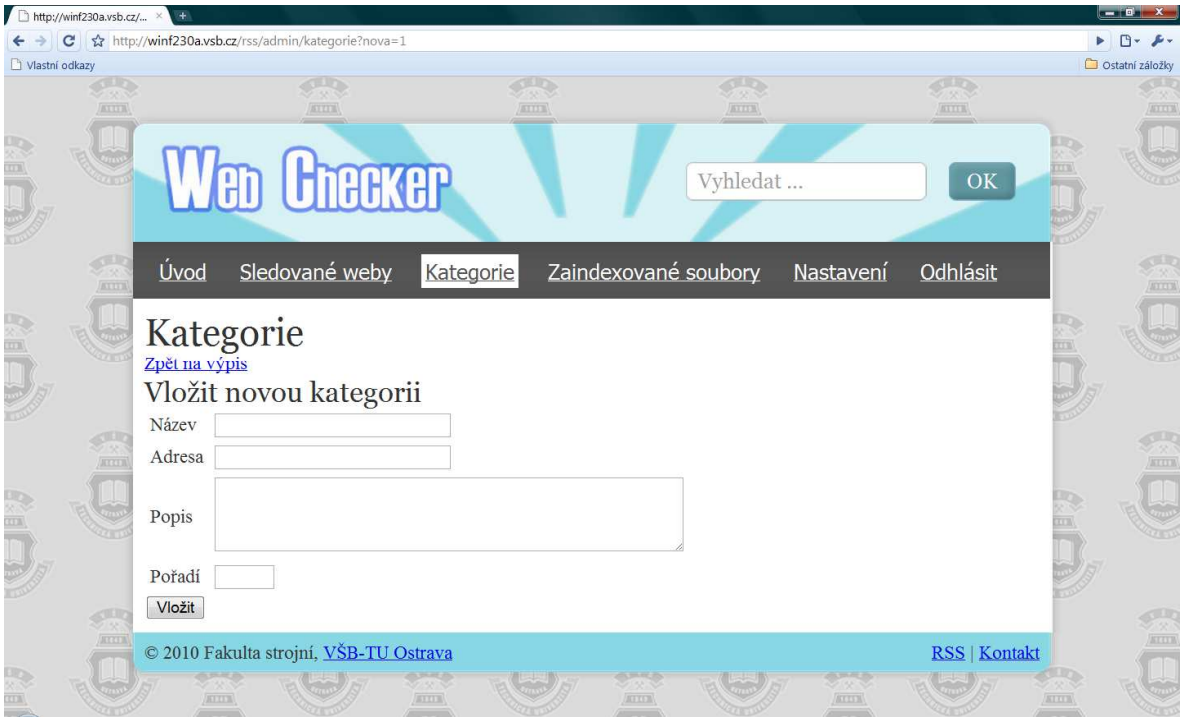
Kategorie

Druhou důležitou částí administrace je stránka, kde se nachází kategorie tohoto portálu. Zde je možné přidat novou kategorii nebo upravit již existující. V seznamu kategorií je také možné snadno změnit jejich pořadí, ve kterém se zobrazují v postraním menu.

Pro vložení nové kategorie je třeba kliknout na odkaz „Vložit novou kategorii“. Po té se zobrazí stránka s formulářem, který obsahuje následující položky.

- **Název –** Jedná se o název, který se bude zobrazovat v postraním menu webu a v seznamu kategorií.
- **Adresa –** Stejně jako při přidání sledovaného webu, se ani zde nejedná se o klasickou http adresu, ale v podstatě o název kategorie bez háčků čárek a mezer, který bude využit pro tvorbu celé adresy.
- **Popis –** Popis se využije v tagu description při výpisu novinek dané kategorie. Do budoucna může být využit i na jiných místech webu. Jako například v tagu title u jednotlivých odkazů kategorie jako „vysvětlující bublinka“.
- **Pořadí –** Stejně jako v seznamu všech kategorií tak i zde je možné změnit pořadí, v jakém se bude daná kategorie zobrazovat.

Editaci kategorie je možné provést kliknutím na odkaz Upravit/Smazat v seznamu kategorií.



Obr. 14: Vložení nové kategorie

Zaindexované soubory

Na této stránce se nachází pouze výpis všech zaindexovaných souborů, které nejsou nijak kategorizovány. Je zde pouze výčet toho co robot zaindexoval. Do budoucna je možné tyto soubory provázat a vytvořit jakousi mapu webů, které jsou sledované.

Nastavení

Na této stránce je možné nastavit, jaké typy souborů má robot sledovat. Typy souborů jsou určené podle koncovky v názvu. Do budoucna by bylo dobré obohatit robota o více možností nastavení, které by umožňovali větší efektivitu při indexaci nebo také možnost různého nastavení pro různé weby.

Závěr

Většina webových stránek Vysoké školy báňské je statického rázu. Některé z nich jsou aktualizované jen výjimečně a nepravidelně. Návštěvníci takových stránek nemají důvod se na takovéto stránky vracet. Pokud se ovšem na stránce objeví nějaká novinka je zpravidla poměrně důležitá a není snadné si jí všimnout.

Z toho důvodu jsem vytvořil vyhledávacího robota, který má za úkol zjišťovat změny na webových stránkách kateder a fakult VŠB. Změny, které robot najde, jsou následně publikované na webovém portálu, kde si je může kdokoli přečíst. Na tomto portále je také možné vytvořit si vlastní seznam webů, které chci sledovat. Výsledný výpis je také možné stahovat pomocí RSS kanálu do jakékoli RSS čtečky nebo třeba na úvodní stránku seznam.cz.

Dále jsem realizoval robota pro agregaci již existujících RSS kanálů, který agreguje zajímavé informace z fakultních webů nebo jiných stránek na univerzitě. RSS kanály bývají zpravidla častěji aktualizované než statické stránky, díky tomu se bude ve výsledném RSS kanálu objevovat často nový obsah. Výsledný výpis je možné sledovat stejným způsobem jako výpis změn.

Výhodou tohoto řešení je sloučení informací z více webů na jednom místě a tak umožnit nalezení požadované informace v přehledném seznamu naposledy přidaných nebo změněných stránek. Studenti VŠB pak už nebudou muset sledovat novinky na všech webových stránkách univerzity, ale všechny novinky přijdou za nimi.

Robota, jako takového, je možné ještě zdokonalit. Jednu z možností bych viděl především v jeho možnostech nastavení, jak obecného, tak nastavení pro konkrétní weby, pro které by platily různá pravidla prohledávání. Dále by také bylo možné přidat hodnocení nalezených změn podle odkazů, které na danou stránku vedou, podobně jako si stránky hodnotí webové vyhledávače.

Webový portál je také možné v mnoha ohledech vylepšit. Mohla by se zde například nacházet mapa všech zaindexovaných souborů s jejich vazbami na ostatní webové stránky. Také by bylo možné dodělat uživatelské hodnocení jednotlivých novinek a počítadlo zhlédnutí, což by se mohlo využít pro řazení novinek ve výpisu. Dalším směrem by mohla být integrace do sociálních sítí. Například vytvoření jednoduché aplikace na facebooku, která by informovala uživatele o změnách.

SEZNAM POUŽITÉ LITERATURY

Asleson Ryan, T. Suchta Nathaniel. *Ajax – Vytváříme vysoce interaktivní webové aplikace*. Computer Press, Brno, 2006. ISBN: 80-251-1285-3

Bureš Jiří. *RSS 2.0. Interval.cz* [Online] 16. 09. 2004 [Citace: 5. Ledna 2010.] Dostupná z http://interval.cz/clanky/rss-20/

Bureš Jiří. *RSS? RSS!. Interval.cz* [Online] 6. 03. 2003 [Citace: 5. Ledna 2010.] Dostupná z http://interval.cz/clanky/rss-rss/

Holzner Steven, Šindelář Jan, *RSS: automatické doručování obsahu vašich WWW stránek. 1. vyd.* Brno: Computer Press, 2007. ISBN: 978-80-251-1479-7.

Chow Shu-Wai. *Programujeme mashup aplikace pro Web 2.0 v PHP. 1.vyd.* Brno: Computer Press, 2008. ISBN 978-80-251-2057-6

Hrebenar Jiří, *SimpleXML - jednoduše na XML v PHP 1.díl* [Online] [Citace: 2. Dubna 2010.] Dostupná z http://programovani.blog.zive.cz/2009/12/simplexml-jednoduse-na-xml-v-php-1dil/

Krug Steve. *Webdesign – Nenuťte uživatele přemýšlet*. Computer Press, Brno, 2006. ISBN: 80-251-1291-8

Kosek Jiří *Využití XML při tvorbě webových aplikací v PHP* [Online] [Citace: 31. Března 2010.] Dostupná z http://www.kosek.cz/vyuka/4iz248/slides/frames.html

Kosek Jiří, *XML pro každého*, Grada Publishing 2000. ISBN: ISBN 80-7169-860-1

Lavin Peter. *PHP - objektově orientované: koncepty, techniky a kód. 1. vyd.* Praha: GRADA Publishing. 2009. ISBN: 978-80-247-2137-8

PHP:cURL – manual [Online] [Citace: 11. Dubna 2010.] Dostupná z http://php.ftp.cvut.cz/manual/en/book.curl.php

PHP: DOM – manual [Online] [Citace: 11. Dubna 2010.] Dostupná z http://php.ftp.cvut.cz/manual/en/book.dom.php

RSS 1.0 *RDF Site Summary (RSS) 1.0* [Online] [Citace: 5. Ledna 2010.] Dostupná z http://web.resource.org/rss/1.0/spec

RSS History *(RSS 2.0 at Harvard Law)* [Online] 6. 4. 2004 [Citace: 5. Ledna 2010.] Dostupná z http://cyber.law.harvard.edu/rss/rssVersionHistory.html

Ullman Larry. *PHP a MySQL*. Computer Press, Brno, 2004. ISBN: 80-251-0063-4

Wikipedia Atom *Atom - Wikipedie, otevřená encyklopedie (standard)* [Online] [Citace: 5. Ledna 2010.] Dostupná z http://cs.wikipedia.org/wiki/Atom_%28standard%29

Wikipedia Document Object Model [Online] [Citace: 11. Dubna 2010.] Dostupná z http://cs.wikipedia.org/wiki/Document_Object_Model

Wikipedia Levenshtein distance - Wikipedia, the free encyclopedia [Online] [Citace: 4. Dubna 2010.] Dostupná z http://en.wikipedia.org/wiki/Levenshtein_distance

Wikipedia RSS *RSS - Wikipedie, otevřená encyklopedie* [Online] [Citace: 5. Ledna 2010.] Dostupná z http://cs.wikipedia.org/wiki/RSS